UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/687,248	10/16/2003	Jimmie Earl DeWitt JR.	AUS920030547US1	9397

35525 7590 05/08/2006

IBM CORP (YA)
C/O YEE & ASSOCIATES PC
P.O. BOX 802333
DALLAS, TX 75380

EXAMINER	
LEE, CHRISTOPHER E	

ART UNIT	PAPER NUMBER
2112	

DATE MAILED: 05/08/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/687,248	DEWITT ET AL.	
	Examiner	Art Unit	
	Christopher E. Lee	2112	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 April 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-6,8,10-15 and 17-22* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-6,8,10-15 and 17-22* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *1/30/06, 2/14/06*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Receipt Acknowledgement

1. Receipt is acknowledged of the Amendment filed on 6th of April 2006. Claims 1, 2, 6, 8, 10, 11, 15, and 17-20 have been amended; claims 7, 9, and 16 have been canceled; and no claim has been newly added since the Non-Final Office Action was mailed on 6th of January 2006. Currently, claims 1-6, 8, 10-15, and 17-22 are pending in this Application.

Double Patenting

2. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970);and, *In re Thorington,* 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

3. Claims 1, 3-6, 8, 10, 12-15, 17, 18, and 20-22 are provisionally rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-5, 9, 11-15, and 17-20 of copending Application No. 10/704,117 in view of Smolders [US 6,253,338 B1]. Although the conflicting claims are not identical, they are not patentably distinct from each other because the instant application is anticipated by the copending application in view of Smolders in that the copending application in view of Smolders suggests obviously all the limitations of the instant application. Therefore, the instant application is not patentably distinct from the earlier filed claims of the copending application in view of Smolders and as such is unpatentable for obvious-type double patenting.

This is a provisional obviousness-type double patenting rejection because the conflicting claims have not in fact been patented.

In regard to claim 1, the copending application discloses all the same limitations as the limitations of the claim 1 in the copending application with the exception of the limitations "said processor being in the data processing system, said indicator indicates enabling a mode of operation in which interrupts are to be generated; and if the initial instruction indicates enabling a mode of operation in which interrupts are to be generated, receiving a subsequent instruction."

However, Smolders discloses a method in a data processing system for processing instructions (i.e., method within a data processing system for counting various events from a running program; See Abstract), wherein responsive to receiving an initial instruction (i.e., instruction for setting a specified branch trace enable bit; See col. 3, lines 58-61) for execution in an instruction cache (i.e., L1 Cache 66 of Fig. 2) in a processor (i.e., Processor 12 of Fig. 2) in the data processing system (i.e., Data Processing System 10 in Fig. 1; See col. 3, lines 4-13), determining whether the initial instruction (i.e., said instruction for setting a specified branch trace enable bit 80 in the machine state register 76 in Fig. 2) indicates enabling a mode of operation (i.e., branch tracing mode enabled by setting BE bit in MSR) in which interrupts (i.e., trace interrupts) are to be generated (See col. 3, line 58 through col. 4, line 11); if the initial instruction indicates enabling a mode of operation in which interrupts are to be generated (i.e., said initial instruction has set BE), receiving a subsequent instruction (i.e., instructions flow) for determining whether the subsequent instruction is of a type selected for analysis (i.e., checking for Branch instructions; See col. 3, line 67 through col. 4, line 6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said method, as disclosed by Smolders, in said data processing

system, as disclosed by the copending application, for the advantage of providing an improved method for tracing hardware counters by way of an interruption without introducing any overhead or modifying the code (See Smolders, col. 1, lines 64-67).

In regard to claim 3, the copending application teaches all the same limitations as the limitations of the claim 2 in the copending application.

In regard to claim 4, the copending application teaches all the same limitations as the limitations of the claim 3 in the copending application.

In regard to claim 5, the copending application teaches all the same limitations as the limitations of the claim 4 in the copending application.

In regard to claim 6, the copending application teaches all the same limitations as the limitations of the claim 5 in the copending application.

In regard to claim 10, the copending application discloses all the same limitations as the limitations of the claim 11 in the copending application with the exception of the limitation "said processor being in the data processing system, said indicator indicates enabling a mode of operation in which interrupts are to be generated; and receiving means for receiving a subsequent instruction if the initial instruction indicates enabling a mode of operation in which interrupts are to be generated."

However, Smolders discloses an apparatus for processing instructions (i.e., a data processing system for counting various events from a running program; See Abstract), wherein first determining means (i.e., means for checking Machine State Register 76 of Fig. 2), responsive to receiving an initial instruction (i.e., instruction for setting a specified branch trace enable bit; See col. 3, lines 58-61) for execution in an instruction cache (i.e., L1 Cache 66 of Fig. 2) in a processor (i.e., Processor 12 of Fig. 2) in the data processing system (i.e., Data Processing System 10 in Fig. 1; See col. 3, lines 4-13), for determining whether the initial

instruction (i.e., said instruction for setting a specified branch trace enable bit 80 in the machine state register 76 in Fig. 2) indicates enabling a mode of operation (i.e., branch tracing mode enabled by setting BE bit in MSR) in which interrupts (i.e., trace interrupts) are to be generated (See col. 3, line 58 through col. 4, line 11); and receiving means (i.e., means for receiving flowing instructions into Instruction flow unit 26 in Fig. 2) for receiving a subsequent instruction (i.e., instructions flow) if the initial instruction indicates enabling a mode of operation in which interrupts are to be generated (i.e., if said initial instruction has set BE; See col. 3, line 67 through col. 4, line 6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said apparatus, as disclosed by Smolders, in said data processing system, as disclosed by the copending application, for the advantage of providing an improved method for tracing hardware counters by way of an interruption without introducing any overhead or modifying the code (See Smolders, col. 1, lines 64-67).

In regard to claim 12, the copending application teaches all the same limitations as the limitations of the claim 12 in the copending application.

In regard to claims 8 and 17, the copending application teaches all the same limitations as the limitations of the claim 9 in the copending application.

In regard to claim 18, the copending application discloses all the same limitations as the limitations of the claim 17 in the copending application with the exception of the limitation "said processor being in the data processing system, said indicator indicates enabling a mode of operation in which interrupts are to be generated; and if the initial instruction indicates enabling a mode of operation in which interrupts are to be generated, second instructions for receiving a subsequent instruction."

However, Smolders discloses an apparatus for processing instructions (i.e., a data processing system for counting various events from a running program; See Abstract), wherein first instructions (i.e., instructions for checking Machine State Register 76 of Fig. 2), responsive to receiving an initial instruction (i.e., instructions for setting a specified branch trace enable bit; See col. 3, lines 58-61) for execution in an instruction cache (i.e., L1 Cache 66 of Fig. 2) in a processor (i.e., Processor 12 of Fig. 2) in the data processing system (i.e., Data Processing System 10 in Fig. 1; See col. 3, lines 4-13), for determining whether the initial instruction (i.e., said instruction for setting a specified branch trace enable bit 80 in the machine state register 76 in Fig. 2) indicates enabling a mode of operation (i.e., branch tracing mode enabled by setting BE bit in MSR) in which interrupts (i.e., trace interrupts) are to be generated (See col. 3, line 58 through col. 4, line 11); and if the initial instruction indicates enabling a mode of operation in which interrupts to be generated (i.e., if said initial instruction has set BE), second instructions (i.e., instructions for receiving flowing instructions into Instruction flow unit 26 in Fig. 2) for receiving a subsequent instruction (i.e., instructions flow; See col. 3, line 67 through col. 4, line 6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said instructions, as disclosed by Smolders, in said data processing system, as disclosed by the copending application, for the advantage of providing an improved method for tracing hardware counters by way of an interruption without introducing any overhead or modifying the code (See Smolders, col. 1, lines 64-67).

In regard to claim 20, the copending application teaches all the same limitations as the limitations of the claim 18 in the copending application.

In regard to claim 21, the copending application teaches all the same limitations as the limitations of the claim 19 in the copending application.

In regard to claim 22, the copending application teaches all the same limitations as the limitations of the claim 20 in the copending application.

4. The mapping of the rejected claims in the instant application to the copending application is as follows.

Instant Application 10/687,248	copending Application No. 10/704,117
1	1 + Smolders
3	2
4	3
5	4
6	5
8, 17	9
10	11 + Smolders
12	12
13	13
14	14
15	15
18	17 + Smolders
20	18
21	19
22	20

Claim Objections

5. The claim 18 recites the subject matter "the data processing system" in line 4. However, it has not been specifically clarified in the claim 18. Therefore, the Examiner presumes that the term "the data processing system" could be considered as --a data processing system-- in light of the specification since it is not defined in the claim.

Claim Rejections - 35 USC § 102

6. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

7. Claims 1, 3-6, 8, 10, 12-15, 17, 18, and 20-22 are rejected under 35 U.S.C. 102(b) as being anticipated by Smolders [US 6,253,338 B1].

Referring to claim 1, Smolders discloses a method in a data processing system for processing instructions (i.e., method within a data processing system for counting various events from a running program; See Abstract), the method comprising:

- responsive to receiving an initial instruction (i.e., instruction for setting a specified branch trace enable bit 80 in the machine state register 76 in Fig. 2; See col. 3, lines 58-61) for execution in an instruction cache (i.e., L1 Cache 66 of Fig. 2) in a processor (i.e., Processor 12 of Fig. 2) in the data processing system (i.e., Data Processing System 10 in Fig. 1; See col. 3, lines 4-13), determining whether the initial instruction (i.e., said instruction for setting said specified branch trace enable bit) indicates enabling a mode of operation (i.e., branch tracing mode enabled by setting BE bit in MSR) in which interrupts (i.e., trace interrupts) are to be generated (See col. 3, line 58 through col. 4, line 11);
- if the initial instruction indicates enabling a mode of operation in which interrupts are to be generated (i.e., said initial instruction has set BE), receiving a subsequent instruction (i.e., receiving instructions flow into Instruction flow unit 26 in Fig. 2; See col. 3, lines 58-61);
- determining whether the subsequent instruction is of a type selected for analysis (i.e., checking for Branch instructions; See col. 3, line 67 through col. 4, line 6); and
- generating an interrupt (i.e., trace interrupt) if the subsequent instruction is determined to be of a type selected for analysis (i.e., check if Branch instruction; See col. 3, lines 58-61).

Referring to claim 3, Smolders teaches the generating step (See col. 3, lines 58-61) comprising:

- sending a signal (i.e., trace interrupt signal after Branch instruction at Step 30 in Fig. 3) from an instruction cache (i.e., Instruction flow unit 26, Execution units 28, and L1 Cache 66 in.Fig. 2) to an interrupt unit (i.e., Performance monitor 24 of Fig. 2) in the processor (i.e., Processor 12 of Fig. 2; See col. 4, lines 21-26); and
- processing the interrupt (i.e., branch tracing interrupt) in the interrupt unit (i.e., said Performance monitor; in fact, counter level tracing tool 31 in Fig. 3 performs handling said branch tracing interrupt) in response to receiving the signal at the interrupt unit (See Fig. 3 and col. 4, lines 12+).

Referring to claim 4, Smolders teaches the processing step (See Fig. 3 and col. 4, lines 12+) including:

- executing code associated with the interrupt (i.e., performing interrupt handling for branch tracing interrupt; See col. 3, lines 55-61; wherein in fact, the interrupt handler code starts executing and the instruction flow unit is programmed to generate a trace interrupt after each branch inherently anticipates the claimed limitation "executing code associated with the interrupt").

Referring to claim 5, Smolders teaches

- the code records cache misses by a functional unit (i.e., load/store unit) attempting to access instructions in a cache (i.e., L2 Cache; See col. 3, lines 43-44).

Referring to claim 6, Smolders teaches

- the code counts a number of times (i.e., counting selected events, e.g., branch tracing interrupt) the subsequent instruction of the type selected for analysis (i.e., Branch instruction) has been executed (See col. 4, lines 6-11).

Referring to claim 8, Smolders teaches

- the subsequent instruction of the type selected for analysis is a branch instruction (See col. 3, line 67 through col. 4, line 6).

Referring to claim 10, Smolders discloses a data processing system for processing instructions (i.e., a data processing system for counting various events from a running program; See Abstract), the data processing system comprising:

- first determining means (i.e., means for checking Machine State Register 76 of Fig. 2), responsive to receiving an initial instruction (i.e., instruction for setting a specified branch trace enable bit 80 in the machine state register 76 in Fig. 2; See col. 3, lines 58-61) for execution in an instruction cache (i.e., L1 Cache 66 of Fig. 2) in a processor (i.e., Processor 12 of Fig. 2) in the data processing system (i.e., Data Processing System 10 in Fig. 1; See col. 3, lines 4-13), determining whether the initial instruction (i.e., said instruction for setting said specified branch trace enable bit) indicates enabling a mode of operation (i.e., branch tracing mode enabled by setting BE bit in MSR) in which interrupts (i.e., trace interrupts) are to be generated (See col. 3, line 58 through col. 4, line 11);
- receiving means (i.e., means for receiving flowing instructions into Instruction flow unit 26 in Fig. 2) for receiving a subsequent instruction (i.e., instructions flow) if the initial

instruction indicates enabling a mode of operation in which interrupts are to be generated (i.e., if said initial instruction has set BE; See col. 3, line 67 through col. 4, line 6);

- second determining means (i.e., means for finding Branch instruction) for determining whether the subsequent instruction is determined to be of a type selected for analysis (i.e., checking for Branch instructions; See col. 3, line 67 through col. 4, line 6); and
- generating means (i.e., Instruction flow unit 26 of Fig. 2) for generating an interrupt (i.e., trace interrupt) if the subsequent instruction is determined to be of a type selected for analysis (i.e., checking if Branch instruction; See col. 3, lines 58-61).

Referring to claim 12, Smolders teaches the generating means (i.e., Instruction flow unit 26 of Fig. 2; See col. 3, lines 58-61) comprising

- means for sending a signal (i.e., means for sending trace interrupt signal after Branch instruction at Step 30 of Fig. 3 within said Instruction flow unit) from an instruction cache (i.e., Instruction flow unit 26, Execution units 28, and L1 Cache 66 in Fig. 2) to an interrupt unit (i.e., Performance monitor 24 of Fig. 2) in the processor (i.e., Processor 12 of Fig. 2; See col. 4, lines 21-26); and
- means for processing the interrupt (i.e., counter level tracing tool 31 processing branch tracing interrupt in Fig. 3) in the interrupt unit (i.e., said Performance monitor; in fact, counter level tracing tool 31 in Fig. 3 performs handling said branch tracing interrupt) in response to receiving the signal at the interrupt unit (See Fig. 3 and col. 4, lines 12+).

Referring to claim 13, Smolders teaches the processing means (i.e., counter level tracing tool 31 processing branch tracing interrupt in Fig. 3) including

- executing means for executing code associated with the interrupt (i.e., means for performing interrupt handling for branch tracing interrupt; See col. 3, lines 55-61, wherein in fact, the interrupt handler code starts executing and the instruction flow unit is programmed to generate a trace interrupt after each branch inherently anticipates the claimed limitation "executing means for executing code associated with the interrupt").

Referring to claim 14, Smolders teaches

- the code records cache misses by a functional unit (i.e., load/store unit) attempting to access instructions in a cache (i.e., L2 Cache; See col. 3, lines 43-44).

Referring to claim 15, Smolders teaches

- the code counts a number of times (i.e., counting selected events, e.g., branch tracing interrupt) the subsequent instruction of the type selected analysis (i.e., Branch instruction) has been executed (See col. 4, lines 6-11).

Referring to claim 17, Smolders teaches

- the subsequent instruction of the type selected for analysis (i.e., Branch instruction) is a branch instruction (See col. 3, line 67 through col. 4, line 6).

Referring to claim 18, Smolders discloses a computer program product in a computer readable medium (See col. 6, lines 14-25) for processing instructions (i.e., a computer program product residing on a computer usable medium for providing counter level tracing for an information handling system; See col. 8, lines 39-41), the computer program product comprising:

- first instructions (i.e., instructions for checking Machine State Register 76 in Fig. 2), responsive to receiving an initial instruction (i.e., instruction for setting a specified branch trace enable bit 80 in the machine state register 76 in Fig. 2; See col. 3, lines 58-61) for execution in an instruction cache (i.e., L1 Cache 66 of Fig. 2) in a processor (i.e., Processor 12 of Fig. 2) in a data processing system (i.e., Data Processing System 10 in Fig. 1; See col. 3, lines 4-13), determining whether the initial instruction (i.e., said instruction for setting said specified branch trace enable bit) indicates enabling a mode of operation (i.e., branch tracing mode enabled by setting BE bit in MSR) in which interrupts (i.e., trace interrupts) are to be generated (See col. 3, line 58 through col. 4, line 11);
- if the initial instruction indicates enabling a mode of operation in which interrupts are to be generated (i.e., if said initial instruction has set BE; See col. 3, line 67 through col. 4, line 6), second instructions (i.e., instructions for receiving flowing instructions into Instruction flow unit 26 in Fig. 2) for receiving a subsequent instruction (i.e., instructions flow; See col. 3, line 67 through col. 4, line 6);
- third instructions (i.e., instructions for finding Branch instruction) for determining whether the subsequent instruction is of a type selected for analysis (i.e., checking for Branch instructions; See col. 3, line 67 through col. 4, line 6); and
- fourth instructions (i.e., instructions for controlling Instruction flow unit 26 in Fig. 2), for generating an interrupt (i.e., trace interrupt) if the subsequent instruction is determined to be of a type selected for analysis (i.e., checking if Branch instruction; See col. 3, lines 58-61).

Referring to claim 20, Smolders teaches the fourth instructions (i.e., instructions for controlling Instruction flow unit 26 in Fig. 2; See col. 3, lines 58-61) comprising

- first sub-instructions for sending a signal (i.e., control instructions for Instruction flow unit 26 of Fig. 2 sending trace interrupt signal after Branch instruction at Step 30 of Fig. 3) from an instruction cache (i.e., Instruction flow unit 26, Execution units 28, and L1 Cache 66 in Fig. 2) to an interrupt unit (i.e., Performance monitor 24 of Fig. 2) in the processor (i.e., Processor 12 of Fig. 2; See col. 4, lines 21-26); and
- second sub-instructions for processing the interrupt (i.e., instructions for counter level tracing tool 31 processing branch tracing interrupt in Fig. 3) in the interrupt unit (i.e., said Performance monitor; in fact, counter level tracing tool 31 in Fig. 3 performs handling said branch tracing interrupt) in response to receiving the signal at the interrupt unit (See Fig. 3 and col. 4, lines 12+).

Referring to claim 21, Smolders teaches the second sub-instructions (i.e., instructions for counter level tracing tool 31 processing branch tracing interrupt in Fig. 3) including

- instructions for executing code associated with the interrupt (i.e., instructions for performing interrupt handling for branch tracing interrupt; See col. 3, lines 55-61, wherein in fact, the interrupt handler code starts executing and the instruction flow unit is programmed to generate a trace interrupt after each branch inherently anticipates the claimed limitation "instructions for executing code associated with the interrupt").

Referring to claim 22, Smolders teaches

- the code records cache misses by a functional unit (i.e., load/store unit) attempting to access instructions in a cache (i.e., L2 Cache; See col. 3, lines 43-44).

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

10. Claims 2, 11, and 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Smolders [US 6,253,338 B1] as applied to claims 1, 3-6, 8, 10, 12-15, 17, 18, and 20-22 above, and further in view of Torrey et al. [US 6,145,123 A; hereinafter Torrey].

Referring to claim 2, Smolders discloses all the limitations of the claim 2, except that does not expressly teach that disabling the mode of operation in which interrupts are to be generated if the subsequent instruction is determined to be of a type to disable the mode of operation in which interrupts are to be generated.

Torrey discloses an information processing system with trace on/off control (See Abstract), wherein

- enabling a mode of operation (i.e., turning on trace operation; See Fig. 5, Steps 530-550, and col. 10, lines 5-11) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19) if an initial instruction is determined

to be of a type to enable a mode of operation in which interrupts are to be generated (i.e., instruction for enabling a debug breakpoint by turning on trace unit ; See col. 5, lines 19-20 and col. 6, lines 56-58); and

- disabling the mode of operation in which interrupts are to be generated (i.e., turning off trace operation; See Fig. 5, Step 570, and col. 10, lines 11-16) if a subsequent instruction is determined to be of a type to disable the mode of operation in which interrupts are to be generated (i.e., instruction for disabling said debug breakpoint by turning off trace unit; See col. 5, lines 19-20 and col. 6, lines 56-58).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said trace on/off control, as disclosed by Torrey, in said data processing system, as disclosed by Smolders, so as to control said mode of operation (i.e., setting/resetting BE bit in MSR for branch trace mode), for the advantage of providing a more specific and relevant trace of instruction being of the certain type (i.e., program operation) for subsequent analysis (See Torrey, col. 3, lines 45-50).

Referring to claim 11, Smolders discloses all the limitations of the claim 11, except that does not expressly teach disabling means for disabling the mode of operation in which interrupts are to be generated if the subsequent instruction is determined to be of a type to disable the mode of operation in which interrupts are to be generated.

Torrey discloses an information processing system with trace on/off control (See Abstract), wherein

- enabling means for enabling a mode of operation (i.e., means for turning on trace operation; See Fig. 5, Steps 530-550, and col. 10, lines 5-11) in which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19) if an

initial instruction is determined to be of a type to enable a mode of operation in which interrupts are to be generated (i.e., instruction for enabling a debug breakpoint by turning on trace unit ; See col. 5, lines 19-20 and col. 6, lines 56-58); and

- disabling means for disabling the mode of operation in which interrupts are to be generated (i.e., means for turning off trace operation; See Fig. 5, Step 570, and col. 10, lines 11-16) if a subsequent instruction is determined to be of a type to disable the mode of operation in which interrupts are to be generated (i.e., instruction for disabling said debug breakpoint by turning off trace unit; See col. 5, lines 19-20 and col. 6, lines 56-58).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have included said means for controlling trace on/off, as disclosed by Torrey, in said data processing system, as disclosed by Smolders, so as to control said mode of operation (i.e., setting/resetting BE bit in MSR for branch trace mode), for the advantage of providing a more specific and relevant trace of instruction being of the certain type (i.e., program operation) for subsequent analysis (See Torrey, col. 3, lines 45-50).

Referring to claim 19, Smolders discloses all the limitations of the claim 19, except that does not expressly teach fifth instructions for disabling the mode of operation in which interrupts are to be generated if the subsequent instruction is determined to be of a type to disable the mode of operation in which interrupts are to be generated.

Torrey discloses an information processing system with trace on/off control (See Abstract), wherein

- fifth instructions (i.e., means for controlling trace on/off) for enabling a mode of operation (i.e., turning on trace operation; See Fig. 5, Steps 530-550, and col. 10, lines 5-11) in

which interrupts (i.e., debug exceptions) are to be generated (See col. 6, line 56 through col. 7, line 19) if an initial instruction is determined to be of a type to enable a mode of operation in which interrupts are to be generated (i.e., instruction for enabling a debug breakpoint by turning on trace unit ; See col. 5, lines 19-20 and col. 6, lines 56-58); and

- said fifth instructions (i.e., said instructions for controlling trace on/off) for disabling the mode of operation in which interrupts are to be generated (i.e., turning off trace operation; See Fig. 5, Step 570, and col. 10, lines 11-16) if a subsequent instruction is determined to be of a type to disable the mode of operation in which interrupts are to be generated (i.e., instruction for disabling said debug breakpoint by turning off trace unit; See col. 5, lines 19-20 and col. 6, lines 56-58).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to have implemented said fifth instructions for controlling trace on/off, as disclosed by Torrey, on said data processing system, as disclosed by Smolders, so as to control said mode of operation (i.e., setting/resetting BE bit in MSR for branch trace mode), for the advantage of providing a more specific and relevant trace of instruction being of the certain type (i.e., program operation) for subsequent analysis (See Torrey, col. 3, lines 45-50).

Response to Arguments

11. Applicants' arguments with respect to all of the pending claims have been considered but are moot in view of the new ground(s) of rejection.

In fact, the Applicants argue with the new issue being drawn to the limitations "if the initial instruction indicates enabling a mode of operation in which interrupts are to be generated, receiving a subsequent instruction," "determining whether the subsequent instruction is of a type selected for analysis," and "generating an interrupt if the subsequent instruction is determined to

be of a type selected for analysis," recited in the amended independent claims 1, 10, and 18, which had not been considered in the prior Office Action, and thus, the Applicants' argument on this point is moot in view of further consideration requirement.

However, the extended scope of the claimed invention in the claims 1, 3-6, 8, 10, 12-15, 17, 18, and 20-22 is still anticipated by Smolders in the record, and the obviousness of the extended scope of the claimed invention in the claims 2, 11, and 19 is still suggested by Smolders and further in view of Torrey in the record (See paragraphs 7 and 10 in the instant Office Action, Claims 1, 3-6, 8, 10, 12-15, 17, 18, and 20-22 rejection under 35 U.S.C. 102(b) as being anticipated by Smolders, and Claims 2, 11, and 19 rejection under 35 U.S.C. 103(a) as being unpatentable over Smolders in view of Torrey).

Conclusion

12. Applicants' amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicants are reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Christopher E. Lee whose telephone number is 571-272-3637. The examiner can normally be reached on 9:30am - 5:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Rehana Perveen can be reached on 571-272-3676. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Christopher E. Lee
Patent Examiner
Art Unit 2112

CEL/